QACTIS Enhancements in TREC QA-2006

P. Schone, G. Ciany, R. Cutts*, P. McNamee†, J. Mayfield†, Tom Smith†*

U.S. Department of Defense
Ft. George G. Meade, MD 20755-6000

ABSTRACT

The QACTIS system has been tested in previous years at the TREC Question Answering Evaluations. This paper describes new enhancements to the system specific to TREC-2006, including basic improvements and thresholding experiments, filtered and Internet-supported pseudo-relevance feedback for information retrieval, and emerging statistics-driven question-answering. For contrast, we also compare our TREC-2006 system performance to that of our top systems from TREC-2004 and TREC-2005 applied to this year's data. Lastly, we analyze evaluator-declared unsupportedness of factoids and nugget decisions of "other" questions to understand major negative changes in performance for these categories over last year.

1. INTRODUCTION

QACTIS (pronounced "cactus"), which breaks out to "Question-Answering for Cross-Lingual Text, Image, and Speech," is a research protoype system being developed by the U.S. Department of Defense. The goals and descriptions of this system are specifically described in past TREC descriptions (see Schone, *et al*., 2004, 2005 in [1], [2]). In this paper, though, we provide a self-contained description of modifications that have been made to the system in 2006. There were three major points of study upon which we conducted research this year: (1) basic improvements to the general processing strategy, (2) information retrieval enhancements as a prefilter, and (3) a move toward integration of more purely-statistical question answering. We describe each of these research avenues in some detail. For the sake of demonstrating these improvements, we evaluate our best systems from TREC-2004 and TREC-2005 on this year's evaluation data as a means of comparison. This comparison does not completely re-create all of the nuances of past-year systems, but we believe it provides an appropriate reflection of system performance over time.

After discussion of the system enhancements, we conduct a post-evaluation analysis of the results from the TREC-2006 evaluation. Our system improved slightly this year in terms of factoid and list answering. However, we experienced 10% and 20% relative losses in system performance due respectively to unsupportedness and inexactness -- numbers which are too large to go without notice. The inexactness losses seem high but hopefully such degradation has been uniformly observed across systems. On the other hand, the unsupportedness is more suspicious. Unlike many other systems, which use the Internet as a pre-mechanism for finding answers and thereby have a chance of recalling the wrong file, our system solely draws its answers from TREC documents and does not mine the Web for answers. We conducted a number of post-evaluation experiments. One of these attempted to make a determination as to whether the unsupported labels were justified. We found through this analysis no systematic biases. Along a similar vein, at TREC-2005, QACTIS's "Other" answerer received the highest score, whereas this year, it suffered a 40% degradation in overall score. We conducted a study to understand this degradation. This evaluation also eliminated concerns about potential assessment problems.

2. CY2006 SYSTEM ENHANCEMENTS

In 2006, there were a number of new avenues of research on QACTIS. As mentioned earlier, these fall into three main directions. Specifically, these involved improvements to the base system, information retrieval enhancements for preselection of appropriate documents, and, lastly, a process to move away from symbolic processing to a more statistical system. We discuss each of these in turn.

2.1 General System Improvements

2.1.1. Overcoming Competition-level Holes

One major modification was designed to overcome problems that only arise with the appearance of fresh data --

* Dragon Development Corporation, Columbia, MD
*Henggeler Computer Consultants, Columbia, MD
†Johns Hopkins Applies Physics Laboratory, Laurel MD

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
2006		**00-00-2006 to 00-00-2006**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
QACTIS Enhancements in TREC QA-2006	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **U.S. Department of Defense,Fort George G. Meade,MD,20755-6000**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**		**9**	

problems which unfortunately only really occur during competitions. In TREC2005, we had noticed that there were a number of questions which our parser failed to properly handle; there were other questions for which the system did not know what kind of information it was seeking; there were questions that were so long that our NIL-tagger generated inadvertent false positives; and lastly, there were many QACTIS-provided answers (as much as 20% relative) marked inexact.

The first two of these were readily solved. We ensured that the system always parsed any previously-unprocessed documents and that it handled these properly at run time. We also attempted to require that any factoid question whose target answer form was unknown would at least return an entity type. Also, we sought to prevent non-responses for other-style questions by requiring the system to revisit the question in a new way if an earlier stage failed to provide a response. These changes were important for yielding a more robust question answerer, and it is possible that as much as 0.5-1% of the factoid score improvements is attributable to these fixes (particularly the parsing fixes).

Perhaps the most dramatic change to handling the problem of the unforeseen was to change the system's scoring metric to take length of question into consideration. Our system attempts to provide a probability-like estimate of the likelihood that some answer is legitimate given the question. All the non-stopwords of the question are important in the question-answering process, so longer questions will naturally have lower estimated probabilities than shorter questions. Nevertheless, it had previously been our policy to use a threshold as a means of estimating whether an answer should be reported as a NIL. This meant that long questions were more likely to erroneously report NIL as their answers than short questions. Through least squares fit, we determined that the probability scores were approximately proportional to $0.01^{\text{length\_in\_words}}$ for questions with at least three words. Therefore, we multiplied the scores of such questions by $100^{(\text{length\_in\_words-3})}$. We conducted an exhaustive search and determined an optimal new NIL threshold (of 10^{-12}). In our developments, this augmentation to the weight seemed to have only limited effect on score but prevented accidental discarding of legitimate answers.

The last challenge was to overcome the large number of answers that QACTIS produced that were being identified as inexact in past years. Our system might identify the appropriate last name of a person that should be reported as an answer, or it might identify the city without the state. We had previously built name and anaphora resolution into our system which we had not been using, and we experimented with various settings of these components to see if we might get some additional gains, but we were unsuccessful. We reasoned that use of a more recently-built content extractor with such resolution embedded could be especially beneficial. BBN was able to generate output for us from their SERIF [3] engine, and we began work to incorporate this information into an exactness filter. Unfortunately, we were not able to make use of this information prior to the evaluation. Ultimately, the only additional resolution that we could incorporate into the system by the time of the competition was to get the base system to augment city names with their corresponding state names when such information was present in the data.

2.1.2. What is the Actual Information Need?

Based on evaluations over past years, we noted that QACTIS produced erroneous answes for questions about court decisions, court cases, ranks, ball teams, scores, campuses, manufacturers, and, in some cases, titles of works of art. These problems were due largely to issues of either underspecificity or to providing a hyponym for a concept rather than a required instance of that concept.

With regard to underspecificity, "teams" provide a great example. If a question were of the form "What team won Super Bowl ...," it is clear to a human that the team that is being referenced is an NFL football team. Instead, if "World Cup" replaced "Super Bowl," the team should be a soccer team. Formerly, the system would seek out any kind of team for the appropriate response -- a problem of underspecificity. To avoid this problem, we encoded knowledge into the system to help it better be able to reach the correct level of specificity particularly with teams. Likewise, in TREC-2005, the system was prepared to identify titles of works of art, but whether or not it could do it was subject to the way the question was posed. We tried to incorporate more generality into its ability to recognize when such a work of art was being requested. We have not removed all problems with this underspecificity, so this will be continued work as we prepare toward TREC-2007.

The hyponym/instance-of issue is likewise a prominent problem in the system. If the system were to see a question "What was the court decision in..." or "what was the score..." the system would think that it was looking for some hyponym of "court decision" and "score" rather than a particular verdict (guilty, innocent) or a numeric value, respectively. We implemented a number of specialized functions to tackle rarer-occuring questions such as these and ensure that the appropriate type of information was provided to the user.

2.1.3. Missing Information Needs: Auto-hypernymy

There are related problems to looking for either hyponyms or instances of classes which are due to lack of world knowledge in some areas. For systems that use the

Internet to provide them potential answers, they get around the problem of missing world information. Our base system is, for the most part, self-contained and we do not currently make direct use of the Web. Therefore, we need to ingest resources to support our question-answering. In past years, we have made use of WordNet[4] and a knowledge source we had previously developed called SemanticForests [5], plus we had targeted specific categories of questions and derived large inventories of potential concepts under those categories through the use of Wikipedia [6]. This year, however, we tried to grow our world knowledge to much more than hundreds of categories and instead try to (a) ingest much or all of Wikipedia's taxonomic structure, and (b) automatically induce taxonomic structure on the fly.

For the first effort, we downloaded the entire English component of Wikipedia and distilled out all of its lists. Then we developed code that could turn those lists into a structure akin to the taxonomic structure required for ingestion by our system. Time constraints limited our ability to do this in a flawless fashion (and revisiting this issue is certainly in order for the future).

In addition to the use of this Wiki-generated taxonomic structure, we also experimented with hypernym induction as a means of finding still more information. In 2004, Snow, et al [7] described an interesting method for hypernym induction that was based on supervised machine learning using features derived from dependency parses. Once trained, the learner can be applied to unannotated corpora to identify new hypenym pairs. Snow provided us with his code, and we began investigating this technology and extending its scalability for application to our question-answering problem.

We used these two new datasets to augment the data that we previously had (and which we had been growing by hand throughout the year) to see if these approaches would yield system improvements. We created two variant taxonomic dictionaries of different sizes and plugged them into the existing system. In Table 1, we illustrate the results of these variants as compared with our baseline system that makes use of largely hand-assembled data. (Note that the scores listed are the number of number one answers (#1) and the F-score for lists on past TREC sets with question identifiers in the specified ranges. In these evaluations, also, the system judgments are automatic, and the judgments do not count off for unsupportedness nor for temporal incorrectness. Moreover, the evaluation counts as correct any outputs which have exact answers embedded therein, but which may not truly be exact. For example, "Coach Harold Solomon" would be scored as correct even if the exact form should only be "Harold Solomon.") The table illustrates a disappointing result: that despite this interesting effort, the taxonomy-growing approach as it currently stands yielded slightly negative results for factoid answering and Variant1 actually yielded significant losses in list performance. Needless to say, we chose to not select these updated data sources for use in our actual evaluation systems. This will be a subject of study for the future.

Table 1: Wikipedia Ingestion

QA Set	Baseline		Variant1		Variant2	
	#1	List	#1	List	#1	List
201-700	165		164			
894-1393	116		113			
1394-1893	133		133		133	
1894-2393	154	.165	152	.137	152	.169
1.1-65.5	72	.197	71	.130	71	.188
66.1-140.5	121	.163	124	.109	121	.157

2.1.4. Longer Term Attempts

There were two other areas of research on QACTIS which we undertook with intentions of incorporating by the time of the evaluation but which required more effort than expected to make ready on time. These are mentioned only briefly for the sake of completeness. One of these areas dealt with morphological information and the other with multitiered processing.

With morphology, our system attempts to crudely generate directed conflation sets for all of the words of the question and for the documents which hopefully contain the answers. A number of questions have been answered incorrectly due to incorrect morphological information related often to word sense. We therefore began what turned out to be a significant effort to convert the way the system did morphological processing to one that would also make use of the part of speech in its stemming process. We hope that this information will strengthen the system at a later point even though it is currently not embedded in the processing.

Another effort which we were not able to finish attempted to convert QACTIS's current process from one that fuses all forms of annotation (named entity, parsing, etc.) from a single stream into one where each stream can be accessed independently. This would allow the system to derive an answer from one stream even if another stream would have yielded some other interpretation. This notion has potentially very positive gains, but we are currently at some distance away from knowing its long-term benefits.

Although the full-blown morphology revision was not incorporated by the time of evaluation, we were able to incorporate a weaker effort regarding morphology with regard to list-processing. The QACTIS system has been developed primarily with a focus on factoid answering, but morphological structure of questions was not well addressed for tackling lists. We therefore did work on

beefing the system up in terms of its list-handling capability and, in the long run, this effort proved to be quite useful in that our list-answering capability on the whole improved substantially.

2.2 Retrieving Documents

Like many other question-answering systems, QACTIS begins its question-answering phase by first attempting to find documents on the same subject We had used the Lemur system [8], version 2.2, since TREC13, and have found its results to be satisfactory. In fact, at TREC14, using this system out of the box yielded one of the top IR systems. We experimented with new versions of Lemur, but were not able to get any better results for QA.

Even still, when we look at the results of our question-answerer, we see that it has a less-than-perfect upper bound due to limitations in information retrieval. If we could enhance our ability to identify appropriate documents, we would likely have a higher performance and a higher upper bound on performance. We set out to improve our ability to preselect documents which would hopefully contain the desired answers. We experimented with two approaches. The first of these was an approach which identified key phrases from the question and tried to ensure that the returned documents actually contained those phrases. We will call this process phrase-based filtering of information retrieval. The second process used the Web as a mechanism for pseudo-relevance feedback. We discuss each of these techniques.

2.2.1 Phrase-based IR Filtering

By the time we had competed our system in the TREC-2005 competition, the base system as applied to one of the older TREC collections (the 1894 question series) was getting mean reciprocal ranks of about 40%. Upon examination of the documents returned by the IR component, it was discovered that a large number of irrelevant documents were being returned. One reason for this was that peoples' names, such as 'Virginia Woolf', were broken into two separate query terms. The resultant documents returned some that contained 'Virginia Woolf' as well as some that related to a 'Bob Woolf' who lived in 'Virginia'. A question pertaining to 'Ozzy Osborne' also returned a document containing a reference to a woman who owned a dog named Ozzy which bit a 'Mrs. Osborne' on the wrist.

Further analysis of the IR set showed that the top 10 documents for each question contained the correct answer 55% of the time; the top 30 documents contained the correct answer 67% of the time. (The numbers were determined by comparing the document list returned by the IR system to the list of correct documents for each question as provided by the TREC competition committee.) The IR system for the 2005 data set was much better-- the top 10 documents contained the correct answer 70% of the time while the top 30 documents contained the correct answer 80% of the time.

A pseudo IR set was built for the 1894 series using the answer set provided by TREC - we referred to this set as a 'perfect IR' set. The number of correct answers provided by the base system when this data set was used was ~60% -- an absolute gain of 20% just by removing irrelevant documents.

An additional step was added to the overall system that attempted to filter the IR using the named entities present in the question. This list also includes dates, titles, and anything in quotes. This process did provide an increase in scores as long as it was not overly aggressive in filtering out too many documents.

Further attempts were made to include multi-word terms and low-frequency words (words in the question which had a lower frequency of occurrence in the overall corpus) as filter terms, but there was not enough time to adequately analyze the effect. Additional parameters such as how many of the top 1000 documents should we examine, how many documents should we retain and how many documents from the original IR should we keep by default also had to be factored in to the result.

By the time of the TREC-2006 evaluation, it was determined that no more than the top 50 documents should be examined. There was no difference in our system in examining 50 or 75 documents. 100 documents degraded the overall system performance. There was also a significant boost in looking at 50 documents as opposed to just 30. Also, because of list questions, it was determined arbitrarily that at least 10 documents should be retained. Since our IR system showed the top 5 docs for each question to be relevant about 60% of the time, we decided to keep the top 5 documents as a matter of course.

We ran our phrase-based filtering on all of the collections at our disposal on the day before the TREC-2006 evaluation. Table 2 illustrates these results (whose scoring follows the paradigm mentioned in Table 1). As can be seen, this approach affords small (2.6% relative) but positive improvements in the overall system performance.

Table 2: Filtered IR DevSet Improvements

QA Set	Baseline			w/ Filtered IR			Diff in #1s
	#1	Mrr	List				
201-700	165	.432		172	.446		+7
894-1393	116	.351		118	.348		+2
1394-1893	133	.393		138	.402		+5
1894-2393	154	.431	.165	158	.444	.176	+4
1.1-65.5	72	.395	.197	71	.402	.197	-1
66.1-140.5	121	.445	.163	127	.456	.162	+6

2.2.2. Google-Enhanced Information Retrieval

The second approach to prefiltering was a multistep technique that took advantage of the Internet without actually trying to mine answers from it. This is a process which apparently has been used by other TREC-QA participants in past years.

To improve our document retrieval phase, we used Google to select augmentation terms for each question. Each question in the test set was converted to a Google query and submitted to Google using the Google API. The top 80 unique snippets returned for each question were used as the augmentation collection. Given a question, we counted the number of times each snippet word co-occurred with a question word in the snippet sentence. These sums were multiplied by the inverse document frequency of the term; document frequencies were calculated from the AQUAINT collection. The resulting scores were used to rank the co-occurring terms. The top eight terms were selected as augmentation terms, and were added to the original query with weight 1/3. The resulting queries were then used for document retrieval.

In selecting these parameters, we were faced with many parameter choices for definition of co-occurrence, term weighting, etc. With over a thousand combinations to choose from, it was not practical to run full question-answering tests on each one to select the best. Instead, we used a proxy for question answering performance: the number of words that occur in question answers that were selected as expansion terms by the method. We mined the answer patterns from past TREC QA tracks for words. Each time a method selected one of these words as an expansion term for the training collection, it was given a point. We used the highest scoring method in our TREC-2006 run.

Table 3: Google-Enhanced Improvements

QA Set		BL	Google-Enhanced				
			TF	LS	LT	S	S2
1894-2393	#1	154	*156*	130	*156*	152	***157***
	Mrr	.431	.455	.377	.455	.442	**.462**
	List	.165	.191	.148	**.192**	.179	.191
1.1-65.5	#1	72	*76*	*80*	***81***	*76*	*79*
	Mrr	.395	.434	.405	.443	.434	
	List	.191	.176	.179	.182	.189	**.195**
66.1-140.5	#1	121	*126*	116	*123*	*122*	***130***
	Mrr	.445	.464	.473	.458	.463	.473
	List	.163	.158	***.179***	.159	.163	*.168*

In our developments with this process, we were quite excited about the gains we were seeing with it. We experimented with five different configurations of this process and one process (S2) yielded particularly successful results. In Table 3, as can be seen from the three most recent years of TREC, as compared to the baseline, the S2 system in preliminary tests yielded a 6.4% relative improvement in factoid performance and a 4.1% relative improvement in list performance.

2.2.3. A Word about Coupled Retrieval

One last experiment we tried was the coupling of these two prefilters. The hope was that if each could give an improvement, then perhaps in combination the improvement would increase. Unfortunately, this was not the case. It turns out that the approaches are somewhat at odds with each other. The phrase-filtering approach attempts to ensure that only documents that contain some or all of the important question phrases should be retained, while the Google-assisted approach attempts to look for documents that might have terminology that was not in the original question. If one applies a phrase-based filtering system to documents that have been obtained by the Google-assisted process, the likelihood is that even fewer of the documents than before will actually have the appropriate terminology. We tried several other variations on this theme after the actual TREC submission, but no combination yielded an improvement in overall performance.

2.3. Beginning to Incorporate Statistical QA

In the past few TREC evaluations, there has been an emergence of statistical QA systems which have the property that they learn relations between the posed question, the answer passage, and the actual answer. Then, when a user poses a future question, various answer passages are evaluated using statistical or machine-learning processes to determine how likely they are to contain a needed answer. As a final step, the system must distill out the answer from the existing passage. Statistical learners are particularly appealing in that they hold potential capability of developing language-independent QA. For such capability, one need only provide question-answer pairs for training.

We began in 2005 to develop a statistical QA system. The infrastructure for this system was in place by the time of the TREC-2006 evaluation and the system had begun to be taught how to automatically answer a limited number of questions, so we thought we would couple it with the existing system and allow it to answer those few question structures that it was equipped to address. Since this system is new and emerging, we provide a bit more information about the process and ways that we attempted to exploit the process during 2006.

2.3.1. From Document Selection to Passage Selection

The first step of the process of developing a statistical system was to move from mere document selection to some

form of passage selection. The first 45 documents reported from the Lemur IR system were screened to identify sentences (and sometimes surrounding sentences) which reflected the information from the important noun words of the question. The first 80 sentences that satisfied the criteria were retained and the sentence selection process was terminated. The 45-document and 80-sentence limits were determined to be empirically optimal threshold.

The next goal was to order these sentences by their potential for answering the question. The reordering component was based on support vector machines (SVMs) . The reordering effort was treated as a two-way classification problem -- a customary domain for SVMs. The classifier was based on 26-dimensional feature vectors that were drawn from the data. Examples of these features were: (a) reporting 1.0 if the direct object from the question was in the putative answer sentence and 0.25 otherwise; (b) reporting 1.0 if the direct object from the question was missing but a WordNet synonym is in the putative answer sentence, and 0.25 otherwise; (c) reporting the ratio of question words to putative answer sentence words; and so forth. The classifier was then presented with positive examples of feature vectors drawn from actual answer sentences of past-year TRECs and it was also presented with a comparable number of negative examples drawn from bogus sentences.

From these training examples, the system was taught with quite good accuracy to learn the difference between good question-answering sentences and poor ones. In fact, for questions where the initial IR actually captured relevant documents, the percentage of true answer sentences identified by the SVM on a held out TREC QA collection was: 30% in the top-1 sentence, 43% in the top-2 sentences, 59% in the top-5 sentences, 74% in the top-10 sentences, and 80% in the top-15 sentences. It seemed highly likely that this process could afford a dramatic improvement in overall performance.

2.3.2. Pulling out the Answers from the Sentences

The next issue was to extract the answer from the answer sentences. One strategy was to insert these sentences into the existing question-answerer and hope it could distill out the answer. This process yielded a 20% relative degradation in performance due largely to the fact that the current system requires itself to find all relevant components of questions, whereas the best sentence may have the answer but not all relevant question components (needed for supportedness). Although we will attempt to modify the base system during the remainder of 2006 and 2007 to tackle the problem, there was also a desire to get a fully-statistical QA system.

We have yet to develop a full statistical learning process for finding answers, but we did begin a simple and potentially language-independent process for answering the questions. Using a copy of Wikipedia, we first used named-entity matching and part of speech tagging to see if we could draw out an answer directly from the Wiki pages. Barring this, we looked for redundant but normally rare information from the SVM sentences and, if it existed, this information was returned as the answer.

3. SYSTEM EVALUATIONS

3.1. Description of Results

In TREC-2006, we submitted three runs from among the various configurations at our disposal. All of our runs used the same "Other" processing as in TREC-2005 (except that this system was slightly more robust to failure than last year). Also, in each situation, the system reported the top 20 answers from our factoid system as the "list" response. In terms of factoids, the first of these runs made use of our base engine but its information retrieval phase was prefiltered using phrase-based filtering as mentioned before. The second system replaced phrase-based filtering with our Google-enhanced information retrieval efforts. The third system was the same as the second but whenever the statistical system was deemed itself able to answer the question, it would supplant the original answer with its own. Since the statistical system is in its infancy, there were very few answers that it actually supplanted.

The results of these runs are detailed in Table 4. Under "Factoid," the number of correct answers is listed and is followed by the triple (unsupported,inexact,temporally-incorrect) and by the fraction of first place answers. Under the "List" and "Other" scores are the NIST-reported F-scores. The "All" category is the average of the three preceding columns, which represents the official NIST score. To our surprise, none of these variations provided significantly different results in the "All" category. However, it seems clear that factoids were negatively impacted by Google-enhanced IR as compared to phrase-based filtering, and the opposite is true for Lists.

Table 4: TREC 2006 Performance

Strategy	Factoid	List	Other	All
Phrase-filtered IR + improved QA [#1]	107 (10/20/5) .266	.147	.148	**.185**
Google-enhanced IR, improved QA [#2]	95 (14/22/4) .236	.156	.151	.181
Google-enhanced IR, improved QA, some statistical QA [#3]	96 (14/22/4) .238	.156	.154	.183

3.2. Comparison to Past Years

With these scores seeming to be only marginally better than they were last year, we wanted to determine if there had actually been any true system improvements since last year. We were able to identify our best competition systems from 2004 and 2005 and conduct a small experiment to test for system improvements by applying these past systems to this year's data. Our experiments would solely focus on factoids and we would not identify questions for which an assessor, had he or she seen the output of the older system, may have judged an answer as correct. Likewise, whereas we had some parsing problems in years past, we would allow the system to directly access the new and updated parses of today (since the broken and/or empty parses have long since been removed). It was our expectation that these two oversights would likely balance each other and provide a fairly accurate comparison of past year performance to that of the current year.

Additionally, since past-year systems were not concerned with temporally inaccurate answers, and since "unsupported" is difficult to truly judge without the input of an assessor, we scored the three systems by allowing what TREC-2006 assessors had declared to be "Right," "Locally Correct", and "Correct but Unsupported." The following table provides the performance comparisons.

Table 5: TREC 2006 vs. Past Years

TREC Year	Factoid #R+L+U/"Correct" Score
TREC-2006 System	122 / .303
TREC-2005 System	95 / .236
TREC-2004 System	53/ .132
(TREC-2006 w/o Filter)	(116 / .288)

Table 5 shows a gratifying result. From the first three rows we see that there have been reasonable improvements to our system over the course of the past two years.

The last row of Table 5 is merely informational. Since we were only able to submit three runs to TREC, we were not able to determine the impact of our basic system improvements as opposed to those that were coupled with IR improvements. With the incorporation of the last row, we are able to see that the basic system improvements contributed to about 21 more right answers and that IR contributed to 6 beyond that. We did not run the experiment of enhanced IR without the basic additions to the system, but while we were originally developing the algorithms, this paradigm was tested and it was typically the case that IR improvements and basic improvements had 1-3 correct answers in common.

3.3. Considering the "Other"s

At TREC-2005, the F=0.248 that our system generated was the maximum score for any of the "Other" question answerers. This year, our F-score dropped by an absolute 10% and our position fell to just above the median. If we had made changes to our "Other" answerer, we would have believed that we had simply put forth a poor effort in making changes. On the other hand, since the only change we made was to add a stage which would thrice ensure against empty answers, we had to seek to understand why the performance would have fallen as it had.

At TREC-2004, our first year of participation, our system received a very high score and only 1/4 of the answers were given zero credit (with half of these due to non-responsiveness of our system). In this year, though, half of our answers were given no credit even though our method for "Other"-answering is sort of a "kitchen sink" approach which reports tons of information. We therefore reviewed the first ten of these zero-scored answers to see what would have changed.

Reviewing the "Other" questions is a non-trivial task. The core issue with these is not knowing how the determination is made as to whether something is vital or not. It appeared this year that since there were so many questions asked from each series, the "Other" questions had little information to choose from that was both novel and vital. Even so, there are three situations that arise in giving a system a zero score in such an the evaluation: (a) the QA system did not return items that assessors found to be valuable, (b) the QA system *did* return such items and received no credit, and (c) the QA system produced items that these assessors deemed to be non-vital but other assessors might have been perceived of as nuggets. Since the task is subjective, an evaluation of "c" is not a particularly helpful direction to study. Yet we will touch briefly on the first two of these given our in-depth review of the ten zero-scored answers that we studied. (It should be noted for reference, though, that there is some subjectivity which is inconsistent: in 164.8, credit is given for the fact that Judi Dench was awarded Order of the British Empire, but credit was not given in 153.8 for Alfred Hitchcock's receiving of higher honors as a Knight Commander.)

By far the biggest problem for us with category "a" above was that what were being deemed as nuggets this year were largely less-important pieces of information which surfaced to the top as vitals because more interesting information was posed as questions. If one were to ask a system: "Tell me everything interesting and unique you can about Warren Moon" (Q141.8), one would expect to receive information about his life profile: birth, death (if appropriate), his profession, and other major successes. Since the Q141 series asks about his position on a football team, the college where he played ball, his birth year, his time as a pro bowler, his coaches, and the teams

on which he played, the remaining relevant information must address his successes and possibly his death. Since he has not died, only successes remain. Thus, the fact that he had the third all-time highest career passing yards, and a 15-year football career are note-worthy. Even so, our IR prefilter rejected one of the documents containing one of these items, and the other item appeared in a 17th-place document ... deeper than our Other processing typically goes.

Further reviewing in the "a" arena, we noted that vital nuggets for 152.7 (Mozart) appeared in our 17th and 60th documents; a vital for 163.8 (Hermitage) was not in our top 100; the sole vital for 164.8 was in 47th place; the two vitals from 175.5 were in 18th place and non-top-100; and so forth. The absence of such information in the higher-IR documents was obviously a leading contributor to our reduced scores. In past years, there were fewer questions asked per series, and many of those questions did not focus so much on key events of people and organizations but were focused more on exercising the QA systems. These facts seem to be the primary reason why our "Other" results would be so drastically degraded.

However, there are a few instances of "b" occurring as well. That is, our system reported vital information that was overlooked. In 163.8, our system reported without credit that the Hermitage was "comparable to the Louvre in Paris or the Metropolitan Museum in New York," which was a specified vital nugget. Such issues are less frequent, though, and they are not unexpected given that our system reports tons of information as answers. Furthermore, if the answers we perused are indicative, the issue of vitals not receiving credit would probably contribute less that .05 absolute to the current F-score.

3.4 Unsupportedness

As mentioned previously, the large number of answers from our system that assessors were tagging as "unsupported" seemed somewhat suspicious to us given that our system does not draw its answers from the Web. We sought to review the answers being proposed by our system and determine what the unsupported issues were.

First, based on the cumulative information provided for all 59 competing system runs, we were able to determine that the average run had 12 answers that were declared to be inexact. We looked at our highest-scored factoid run and noted that we had 10 apparently unsupported answers. Although 10 was less than the average, we still wanted to understand the issues. We reviewed each answer and found that *all* the answers were indeed unsupported (and possible inexact as well). The table below summarizes this information: the question number (QID), the answer our system reported, and the reason why the answer was unsupported.

Table 6: Unsupported Answers: Why?

QID	Our Answer [Document]	Reason Unsupported
145.4	february 23, 1999 [NYT19990302.0069]	Needed a conviction date. Document dated 3/2/1999 refers to "last week" which is ambiguous.
154.4	Margot Kidder [APW19981213.1025]	Needed most-frequent actress in Superman. Document states that Kidder starred with Reeve, but nothing about "most"
172.1	Burlington [NYT20000124.0364]	Needed a city and state of company's origin. The answer is inexact, too, but this document only says "based in," not originated in.
182.5	Scotland [APW19990506.0176]	Needed country of Edinburg Fringe. Document discusses politics in Scotland and a "fringe party" -- polysemy problem.
188.1	California [APW19990117.0079]	Needed US state with highest avocado production. California is only mentioned in passing and nothing mentions production rates.
189.7	Edinburgh [NYT20000112.0203]	Needed city of JK Rowling in 2000. Document states that she lived in Edinburgh in 1993. Unclear if she lived there in 2000.
190.1	PITTSBURGH [APW19990615.0036]	Needed city of HJ Heinz. The byline gives Pittsburgh and discusses company profits, but does not explicitly say its base as there.
191.3	Germany [XIE19990620.0031]	Needed country that won first four IRFR World Cups. Document mentions "Germany" and "four" but not that Germany won 4 times.
194.3	six [XIE19960320.0094]	Need number of players at 1996 World Chess Super Tournament. Document mentions 6 players, but wrong tournament and game.
214.6	seven [NYT20000717.0370]	Need number of Miss America pageant judges. Document is on Miss Texas Scholastic Pageant which had 7 judges.

4 FUTURE DIRECTIONS

The future of QACTIS still holds a direction of multilingual and multimedia question-answering as a primary

goal. Yet we anticipate future participation in TREC next year until we have ironed out the wrinkles in our system. Our focus on textual QA for the next year will be to address the issues that have yet to be completed but what were mentioned in this paper, such as improvements and exactness filtering using more modern content extractors, better incorporation of hypernyms, and making improvements to our statistical QA system. We also plan to make our baseline system cleaner and more robust.

5 REFERENCES

[1] Schone, P., Ciany, G., McNamee, P., Kulman, A., Bassi, T. , "Question Answering with QACTIS at TREC 2004" *The 13th Text Retrieval Conference (TREC-2004*), Gaithersburg, MD. NIST Special Publication 500-261,2004.

[2] Schone, P., Ciany, G., Cutts, R.., McNamee, P., Mayfield, J., Smith, T. , "QACTIS-based Question Answering at TREC-2005" The 14th Text Retrieval Conference (TREC-2005), Gaithersburg, MD, ,2005.

[3] BBN's SERIF^{TM} engine.

[4] Miller G. A., Beckwith R., Fellbaum C., Gross D., and Miller K. J. "WordNet: An online lexical database." *International Journal of Lexicography* 3(4): 235-244, 1990.

[5] P. Schone, J. Townsend, C. Olano, T.H. Crystal. "Text Retrieval via Semantic Forests." TREC-6, Gaithersburg, MD. *NIST Special Publication 500-240*, pp. 761-773, 1997

[6] www.wikipedia.org

[7] Snow, R., Jurafsky, D., and Ng, A.Y., "Learning syntactic patterns for automatic hypernym discovery". *NIPS 2004*.

[8] The LEMUR System. URL: http://www-2.cs.cmu.edu/~lemur